IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants	:	Jewett, et al.
Appl. No.	:	09/927,894
Filed	:	August 10, 2001
For	:	ARCHITECTURE FOR PROVIDING BLOCK-LEVEL STORAGE ACCESS OVER A COMPUTER NETWORK
Examiner	:	Sargon N. Nano
Group Art Unit	:	2157

AMENDED APPEAL BRIEF

United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Appellants, Applicants in the above-captioned patent application, appeal the rejection of Claims 1-37 and 54-59 set forth in the Office Action mailed on March 9, 2006 (hereinafter "the current Office Action"). Please charge any additional fees that may be required now or in the future to Deposit Account No. 11-1410.

I. REAL PARTY IN INTEREST

The real party in interest in the present application is 3ware, Inc., which is a subsidiary of Applied Micro Circuits Corporation (AMCC).

II. RELATED PROCEEDINGS

No related appeals, interferences or judicial proceedings are currently pending.

III. STATUS OF CLAIMS

Claims 1-37 and 54-59, which are attached hereto as an appendix, are currently pending in the application and are the subject of this appeal. Claims 38-53 are canceled.

IV. STATUS OF AMENDMENTS

An amendment was filed on June 6, 2006, to correct a typographical error discovered in Claim 28.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The present application includes four independent claims. Each independent claim is summarized below, with citations to corresponding portions of the specification and drawings as required by 37 C.F.R. § 41.37(c)(1)(v). These citations and the associated explanations are provided to illustrate specific examples and embodiments of the claimed subject matter, and do not limit the claims. In the following summaries, the first digit of each reference number identifies the figure in which the referenced item first appears.

Claim 1 is directed to a block-level shared network storage system. The system comprises the following:

- a storage server (104) comprising an array of disk drives (112), and comprising a processor (108) that runs a device driver (206) to provide block-level access to data stored on the array of disk drives (112) (see, e.g., page 5, line 19 to page 6, line 5; and page 6, lines 17-25); and
- a host computer (102) coupled to the storage server (104) by at least one computer network (100) (see, e.g., page 5, line 19 to page 6, line 16);
- wherein the host computer (102) and the storage server (104) perform input/output (I/O) operations over the at least one network (100) using multiple, concurrent logical connections (400 in Fig. 4), each logical connection being between the host computer (102) and the storage server (104) over the at least one computer network (100), such that a first I/O operation is executed over a first logical connection while a second I/O

operation is executed over a second logical connection (see, e.g., page 2, lines 10-19; page 7, lines 18-23; and page 9, lines 9-23).

One embodiment of a storage system as described in Claim 1 is shown in Figs. 1-4, in which the block server 104 represents the "storage server" of the claim. Figure 4 illustrates a configuration in which multiple host computers 102 are connected to multiple storage servers 104. In this illustrated embodiment, the "logical connections" of Claim 1 are in the form of socket connections 400 (see Fig. 4) established over a network 100 or combination of networks. As shown in Fig. 4 and explained at page 2, lines 10-19; page 7, lines 18-23; and page 9, lines 9-23 of the specification, multiple concurrent socket connections 400 can be established between a host computer 102 and a storage server 104, and used in parallel to perform I/O operations such that one I/O operation is performed over a first socket connection while another I/O operation is performed over another I/O operation. As explained at page 7, lines 20-23, if one of these socket connections fails in Applicants' embodiment, I/O requests can continue to be performed over the remaining socket connection or connections.

Claim 19 is directed to a system for storing data for host computers, comprising:

- a plurality of storage servers (104 in Fig. 4) connected to a network (100), each storage server comprising an array of disk drives (112), an array controller (110), and a processor (108) (see, e.g., page 5, lines 8-11; and page 5, line 19 to page 6, line 5);
- a plurality of host computers (102) connected to the network (100) and programmed to store data on the storage servers (104) (see, e.g., page 5, lines 8-11; Fig. 4; and page 9, lines 4-23); and
- at least one switch (308) which interconnects the plurality of storage servers (104) with the plurality of host computers (102) (see, e.g., page 5, lines 15-18; Fig. 3; and page 8, lines 7-14);
- wherein each host computer (102) is programmed to open multiple concurrent socket connections (400) over the network (100) to the storage servers (104) and to perform input/output operations in parallel over the multiple concurrent socket connections (400) (see, e.g., page 2, lines 10-16; page 7, lines 18-23; Fig. 4, and page 9, lines 9-23).

One embodiment of a system as described in Claim 19 is shown in Figs. 1-4, in which the block servers 104 represent the "storage servers" of the claim. As shown in Fig. 4 and explained at page 2, lines 10-19; page 7, lines 18-23; and page 9, lines 9-23 of the specification, a host computer 102 can open multiple concurrent socket connections 400 to the storage servers 104 over a network 100, and can perform I/O operations in parallel over these socket connections. In the scenario depicted in Fig. 4, the uppermost host computer 102 has opened four concurrent socket connections 400 to the storage servers 104, two to block server #1, and two to block server #2.

Claim 28 is directed to a method of performing input/output operations. The method comprises:

- establishing first and second TCP/IP connections between a host computer (102) and a block-level storage server (104) over one or more computer networks (100) (see, e.g., page 2, lines 10-19; page 2, line 30 to page 3, line 5; page 5, lines 11-15; page 7, lines 18-23; Fig. 4; and page 9, lines 9-23);
- performing a first input/output operation over the first TCP/IP connection concurrently with performing a second input/output operation over the second TCP/IP connection, each of said input/output operations comprising a transfer of input/output data between the host computer (102) and the storage server (104) (see, e.g., page 2, lines 10-19; page 2, line 30 to page 3, line 5; page 7, lines 18-23; Fig. 4; page 9, lines 9-23; and page 12, line 20 to page 13, line 9); and
- maintaining the first and second TCP/IP connections in a persistent state such that each TCP/IP connection may be used to perform additional input/output operations (see, e.g., page 2, line 30 to page 3, line 5; page 7, lines 4-10; and page 22, lines 13-18).

One example of a set of components that can be used to perform the method of Claim 28 is shown in Figs. 1-4, in which the block servers 104 represent the "block-level storage servers" of the claim. The "TCP/IP connections" mentioned in the claim are implemented in this embodiment as socket connections 400, depicted in Fig. 4. (As is well known in the art, a TCP/IP connection can, but need not, be implemented as a socket connection, and a socket

connection can but need not be a TCP/IP connection.) As explained at page 2, lines 10-19; page 7, lines 18-23; page 9, lines 9-23; and page 12, line 20 to page 13, line 9, the disclosed system implements a process in which a first I/O operation is performed over a first TCP/IP connection while a second I/O operation is performed over a second TCP/IP connection, with each such I/O operation comprising a transfer of I/O data between a host computer 102 and a storage server 104. As explained at page 2, line 30 to page 3, line 5; page 7, lines 4-10; and page 22, lines 13-18, these TCP connections are preferably maintained in a persistent state such that each TCP/IP connection may be used to perform additional I/O operations.

Claim 34 is directed to a method of executing an input/output (I/O) request received from a user-level process running on a host computer. The method comprises:

- on the host computer (102), dividing the I/O request into multiple constituent I/O operations (see, e.g., page 3, lines 6-13 and page 12, line 20 to page 13, line 16); and
- performing the multiple constituent I/O operations in parallel over multiple, respective logical network connections (400) between the host computer (102) and a target storage server (104) such that I/O data is transferred between the host computer (102) and the storage server (104) over each of the logical network connections (see, e.g., page 3, lines 6-13; page 4, lines 21-23; page 7, lines 18-23; and page 12, line 20 to page 13, line 16).

One example of how the method of Claim 34 may be implemented is provided at page 3, lines 6-13 of the specification. Specifically, when an I/O request from a host process involves the transfer of more than a threshold quantity of data, the host's device driver 204 divides the I/O request into two or more constituent I/O operations. Each such operation is assigned to a different socket connection 400 with the target storage server 104 such that the constituent operations may be performed, and the associated I/O data transferred, in parallel over the network 100. Additional details of a particular embodiment of this feature are provided at Fig. 7 and page 12, line 20 to page 13, line 16.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The sole ground for rejection to be reviewed on appeal is the rejection of Claims 1-37 and 54-59 under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,834,326 ("Wang"). Appellants reserve the right to disqualify Wang as prior art in a subsequent proceeding.

VII. ARGUMENT

For the reasons set forth below, the anticipation rejections of Claims 1-37 and 54-59 are improper and should be reversed. By declining to present separate arguments in connection with certain dependent claims, Appellants do not imply an agreement with, and do not acquiesce in, the Examiner's positions with respect to these claims.

Independent Claim 1

Claim 1 reads as follows, with selected limitations italicized for purposes of discussion:

> 1. A block-level shared network storage system, comprising:
> a storage server comprising an array of disk drives, and comprising a processor that runs a device driver to provide block-level access to data stored on the array of disk drives; and
> a host computer coupled to the storage server by at least one computer network;
> *wherein the host computer and the storage server perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections, each logical connection being between the host computer and the storage server over the at least one computer network, such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection.*

The anticipation rejection of Claim 1 is improper because Wang does not explicitly or inherently disclose the limitations italicized above. *See Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631; 2 USPQ2d 1051, 1053 (Fed. Cir. 1987) ("A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.").

In connection with these limitations, the Examiner points to col. 8, lines 42-54 and Figure 6 of Wang. The text at column 8, lines 42-54 refers to Figure 5, and describes how a single RAID controller can access multiple RAID volumes, and can also access an independent disk drive that is not part of a RAID volume. Figure 6 shows how two RAID controllers can be

interconnected such that the lower-most RAID controller, together with its associated disk drives, appears as a separate disk drive to the upper-most RAID controller. See column 8, lines 56-63.

Even if one of the RAID controllers in Figure 6 is treated as the "storage server" recited in Claim 1, and the other RAID controller is treated as the "host computer," there is still no anticipation.[1] In this regard, Wang does not disclose that these two RAID controllers "perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections... such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection." In addition, the Examiner has made no attempt to show that these limitations are inherent in Wang. See MPEP 2112 ("In relying upon the theory of inherency, the examiner must provide a basis in fact and/or technical reasoning to reasonably support the determination that the allegedly inherent characteristic necessarily flows from the teachings of the applied prior art.") (citation omitted).

Because Wang does not explicitly or inherently disclose all of the limitations of Claim 1, the anticipation rejection of Claim 1 is improper and should be reversed.

Dependent Claims 2-18 and 54

Claims 2-18 and 54 depend from Claim 1. Thus, the rejections of these dependent claims are improper for the reasons explained above for Claim 1. The rejections of these dependent claims are also improper, and should be reversed, for the additional reasons set forth below for specific dependent claims.

Dependent Claim 5

The rejection of dependent Claim 5 is additionally improper because Wang does not disclose the following limitations added by Claim 5: "the host computer is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical connections of said multiple, concurrent logical connections." In connection with these limitations, the Examiner again points to column 8, lines 42-54 of Wang. Neither this nor any other portion of Wang, however, discloses either the

[1] During a telephone interview conducted on April 25, 2006, Applicants' representative asked the Examiner to explain how he is reading col. 8, lines 42-54 and Fig. 6 of Wang as disclosing the italicized limitations. The Examiner was unable to provide such an explanation.

division of an I/O operation into multiple constituent I/O operations, or the performance of multiple constituent I/O operations in parallel over respective logical connections.

Dependent Claim 7

The rejection of dependent Claim 7 is additionally improper because Wang does not disclose the following limitations of Claim 7 "the storage server is configurable to provide multiple storage partitions, each of which may be allocated to a different host computer." In connection with this claim, the Examiner points to column 5, lines 5-24, which describes the RAID-0 standard. Nothing in this description, however, suggests the allocation of different storage partitions provided by a storage server to different host computers.

Dependent Claim 15

The rejection of dependent Claim 15 is additionally improper because Wang does not disclose the following limitations: "the first and second logical connections exist over separate computer networks." In connection with this claim, the Examiner merely points to Figure 6 of Wang, without making any attempt to explain how the drawing discloses the limitations at issue. Indeed, the drawing does not.

Dependent Claim 18

The rejection of dependent Claim 18 is additionally improper because Wang does not disclose the following limitations: "the host computer and the storage server each include two network interfaces that provide redundant network connections between the host computer and the storage server." The portion of Wang cited by the Examiner in connection with this claim, namely col. 10, line 66 to column 11, line 10, simply does not disclose this feature.

Independent Claim 19

Claim 19 reads as follows, with selected limitations italicized:

> 19. A system for storing data for host computers, comprising:
>
> a plurality of storage servers connected to a network, each storage server comprising an array of disk drives, an array controller, and a processor;
>
> a plurality of host computers connected to the network and programmed to store data on the storage servers; and
>
> at least one switch which interconnects the plurality of storage servers with the plurality of host computers;
>
> *wherein each host computer is programmed to open multiple concurrent socket connections over the network to the storage servers and to perform*

input/output operations in parallel over the multiple concurrent socket connections.

The anticipation rejection of independent Claim 19 is improper because Wang does not explicitly or inherently disclose the limitations italicized above. In connection with these limitations, the Examiner again points to col. 8, lines 42-54 and Figure 6 of Wang. Neither these nor any other portions of Wang, however, explicitly or inherently disclose the limitations at issue. In this regard, although Wang discloses the use of sockets, there is nothing in Wang to suggest a host computer that opens, and performs input/output operations in parallel over, multiple concurrent socket connections over a network. This is true even if one of the RAID controllers shown in Figure 6 is treated as a "host computer."

Because Wang does not explicitly or inherently disclose all of the limitations of Claim 19, the anticipation rejection of Claim 19 is improper and should be reversed.

Dependent Claims 20-27, 55 and 56

Claims 20-27, 55 and 56 depend from Claim 19. Thus, the rejections of these claims are improper for the reasons set forth above for Claim 19. The rejections of these dependent claims are also improper, and should be reversed, for the additional reasons set forth below for specific claims.

Dependent Claim 22

The rejection of dependent Claim 22 is additionally improper because Wang does not disclose the following limitations: "wherein at least a first host computer of the plurality of host computers is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical socket connections between the first host computer and a target storage server." The portions of Wang relied on by the Examiner in connection with Claim 22, namely column 8, lines 42-54 and Fig. 6, simply do not disclose this feature.

Dependent Claim 24

The rejection of dependent Claim 24 is additionally improper because Wang does not disclose the following limitations: "wherein a first storage server of the plurality of storage

servers has a first partition which is uniquely assigned to a first host computer of the plurality of host computers such that the first partition appears as a local disk drive to the first host computer." The portions of Wang relied on by the Examiner in connection with Claim 24, namely column 8, lines 56-63 and Figure 6, do not disclose the unique assignment of a partition of a storage server to a particular host computer as claimed.

Dependent Claim 55

The rejection of dependent Claim 55 is additionally improper because Wang does not disclose the following limitations: "a first host computer of said plurality of host computers is programmed to open first and second socket connections over the network to a first storage server of said plurality of storage servers, and to perform a first input/output operation over the first socket connection while performing a second input/output operation over the second socket connection." The portions of Wang relied on by the Examiner, namely column 8, lines 42-54 and Fig. 6, simply do not disclose these limitations.

Independent Claim 28

Claim 28 reads as follows:

> 28. A method of performing input/output operations, comprising:
> establishing first and second TCP/IP connections between a host computer and a block-level storage server over one or more computer networks;
> performing a first input/output operation over the first TCP/IP connection concurrently with performing a second input/output operation over the second TCP/IP connection, each of said input/output operations comprising a transfer of input/output data between the host computer and the storage server; and
> maintaining the first and second TCP/IP connections in a persistent state such that each TCP/IP connection may be used to perform additional input/output operations.

The anticipation rejection of independent Claim 28 is improper because Wang does not explicitly or inherently disclose the following limitations: "performing a first input/output operation over the first TCP/IP connection [between a host computer and a block-level storage server] concurrently with performing a second input/output operation over the second TCP/IP connection [between the host computer and the block-level storage server], each of said input/output operations comprising a transfer of input/output data between the host computer and the storage server." The anticipation rejection is also improper because Wang does not explicitly

or inherently disclose "maintaining the first and second TCP/IP connections in a persistent state such that each TCP/IP connection may be used to perform additional input/output operations."

In connection with the above limitations, the Examiner again cites col. 8, lines 42-54 of Wang. As explained above, the text at column 8, lines 42-54 refers to Figure 5, and describes how a single RAID controller can access multiple RAID volumes, and how this RAID controller can also access an independent disk drive that is not part of a RAID volume. Nothing in this description, or any other portion of Wang, explicitly or inherently discloses the establishment and use of first and second TCP/IP connections as claimed. Indeed, the cited portion of Wang does not even discuss how connections are created and used.

Because Wang does not explicitly or inherently disclose all of the limitations of Claim 28, the anticipation rejection of Claim 28 is improper and should be reversed.

Dependent Claims 29-33 and 57

Claims 29-33 and 57 depend from Claim 28. Thus, the anticipation rejections of these claims are improper for the reasons set forth above for Claim 28. The rejections of these dependent claims are also improper, and should be reversed, for the additional reasons set forth below for specific claims.

Dependent Claim 29

The rejection of dependent Claim 29 is additionally improper because Wang does not disclose the following limitations: "the first and second TCP/IP connections are established over separate computer networks." The portion of Wang cited by the Examiner as disclosing these limitations, namely col. 8, lines 42-54, says nothing to suggest that connections are established over separate networks as claimed.

Dependent Claim 30

The rejection of dependent Claim 30 is additionally improper because Wang does not disclose the following limitations: "the first input/output operation is a first I/O request issued from a first application running on the host computer, and the second input/output operation is a second I/O request issued from a second application running on the host computer." In connection with these limitations, the Examiner points to column 9, lines 13-43 of Wang. While this portion of Wang may refer to multiple applications, it says nothing to suggest that I/O requests issued by these applications are processed in the manner claimed.

Dependent Claim 31

The rejection of dependent Claim 31 is additionally improper because Wang does not disclose the following limitations: "the first and second input/output operations are constituent operations of an I/O request issued by a process running on the host computer, whereby the I/O request is executed in parallel over multiple TCP/IP connections." The portions of Wang cited by the Examiner in connection with Claim 31, namely col. 8, lines 42-54 and Fig. 6, say nothing to suggest the execution of an I/O request in parallel over multiple TCP/IP connections as claimed.

Dependent Claim 32

The rejection of dependent Claim 32 is additionally improper because Wang does not disclose the following limitations: "further comprising establishing a third TCP/IP connection between the host computer and the storage server, and using the third TCP/IP connection to perform an authentication sequence in which the storage server authenticates the host computer." Although the cited portion of Wang (namely column 10, lines 33-47) discusses security generally, it says nothing to suggest the use of a third TCP/IP connection to perform an authentication sequence as claimed.

Independent Claim 34

Independent Claim 34 reads as follows:

> 34. A method of executing an input/output (I/O) request received from a user-level process running on a host computer, comprising:
>
> on the host computer, dividing the I/O request into multiple constituent I/O operations; and
>
> performing the multiple constituent I/O operations in parallel over multiple, respective logical network connections between the host computer and a target storage server such that I/O data is transferred between the host computer and the storage server over each of the logical network connections.

The rejection of Claim 34 is improper because Wang does not explicitly or inherently disclose either "on the host computer, dividing the I/O request into multiple constituent I/O operations," or "performing the multiple constituent I/O operations in parallel over multiple, respective logical network connections between the host computer and a target storage server

such that I/O data is transferred between the host computer and the storage server over each of the logical network connections."

In connection with these limitations, the Examiner relies again on column 8, lines 42-54 of Wang. Nothing at this or any other portion of Wang, however, discloses or suggests dividing an I/O request into multiple constituent I/O operations, or performing these I/O operations in parallel over multiple, respective logical network connections, as claimed.

Because Wang does not explicitly or inherently disclose all of the limitations of Claim 34, the anticipation rejection of Claim 34 is improper and should be reversed.

Dependent Claims 35-37 and 58 and 59

Claims 35-37, 58 and 59 depend from Claim 34. Thus, the anticipation rejections of these claims are improper for the reasons set forth above for Claim 34.

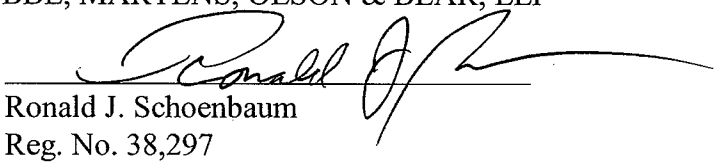
VIII. CONCLUSION

For the reasons set forth above, Appellants respectfully submit that the anticipation rejections of Claims 1-37 and 54-59 are improper and should be reversed.

Respectfully submitted,

KNOBBE, MARTENS, OLSON & BEAR, LLP

Dated: 10-5-06

By: 

Ronald J. Schoenbaum
Reg. No. 38,297
Knobbe, Martens, Olson & Bear, LLP
2040 Main Street, 14th Floor
Irvine, CA 92614
949-721-2950

CLAIMS APPENDIX

1. A block-level shared network storage system, comprising:

a storage server comprising an array of disk drives, and comprising a processor that runs a device driver to provide block-level access to data stored on the array of disk drives; and

a host computer coupled to the storage server by at least one computer network;

wherein the host computer and the storage server perform input/output (I/O) operations over the at least one network using multiple, concurrent logical connections, each logical connection being between the host computer and the storage server over the at least one computer network, such that a first I/O operation is executed over a first logical connection while a second I/O operation is executed over a second logical connection.

2. The network storage system as in Claim 1, wherein each logical connection is a socket connection.

3. The network storage system as in Claim 2, wherein each socket connection is a TCP/IP socket connection.

4. The network storage system as in Claim 1, wherein each logical connection remains persistent over multiple I/O operations performed over that logical connection.

5. The network storage system as in Claim 1, wherein the host computer is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical connections of said multiple, concurrent logical connections.

6. The network storage system as in Claim 1, wherein the storage server provides virtualized block-level storage access to the host computer such that the storage server is treated as local disk drive storage by user-level processes running on the host computer.

7. The network storage system as in Claim 1, wherein the storage server is configurable to provide multiple storage partitions, each of which may be allocated to a different host computer.

8. The network storage system as in Claim 1, wherein the storage server has a first storage partition which is uniquely assigned to the host computer such that the first storage partition appears to user-level processes running on the host computer as a private, local disk drive.

9. The network storage system as in Claim 8, wherein the storage server further has a second storage partition which is uniquely assigned to a second host computer.

10. The network storage system as in Claim 8, wherein the storage server further has a second storage partition which is shared by multiple host computers.

11. The network storage system as in Claim 1, wherein the host computer and the storage server implement an authentication protocol in which the storage server authenticates the host computer before allowing the host computer to perform input/output operations.

12. The network storage system as in Claim 1, wherein the host computer and the storage server implement a discovery protocol in which the storage server notifies the host computer of partitions assigned to the host computer.

13. The network storage system as in Claim 1, wherein at least one of the logical connections is over a general-purpose computer network.

14. The network storage system as in Claim 1, wherein at least one of the logical connections is over an Ethernet network.

15. The network storage system as in Claim 1, wherein the first and second logical connections exist over separate computer networks.

16. The network storage system as in Claim 1, wherein each logical connection exists between a respective reader/writer pair.

17. The network storage system as in Claim 1, wherein the host computer and the storage server are interconnected by at least one switch.

18. The network storage system as in Claim 1, wherein the host computer and the storage server each include two network interfaces that provide redundant network connections between the host computer and the storage server.

19. A system for storing data for host computers, comprising:

a plurality of storage servers connected to a network, each storage server comprising an array of disk drives, an array controller, and a processor;

a plurality of host computers connected to the network and programmed to store data on the storage servers; and

at least one switch which interconnects the plurality of storage servers with the plurality of host computers;

wherein each host computer is programmed to open multiple concurrent socket connections over the network to the storage servers and to perform input/output operations in parallel over the multiple concurrent socket connections.

20. The system of Claim 19, wherein the socket connections are TCP/IP socket connections.

21. The system of Claim 19, wherein each storage server of the plurality of storage servers provides virtualized block-level storage access to the host computers such that the storage servers are treated as local disk drive storage by user-level processes running on the host computers.

22. The system of Claim 19, wherein at least a first host computer of the plurality of host computers is programmed to divide an I/O operation into multiple constituent I/O operations, and to perform the multiple constituent I/O operations in parallel over respective logical socket connections between the first host computer and a target storage server.

23. The system of Claim 19, wherein a first storage server of the plurality of storage servers is configurable to provide multiple, variable-size partitions, each of which may be allocated to a different host computer of the plurality of host computers.

24. The system of Claim 19, wherein a first storage server of the plurality of storage servers has a first partition which is uniquely assigned to a first host computer of the plurality of host computers such that the first partition appears as a local disk drive to the first host computer.

25. The system of Claim 24, wherein the first storage server further has a second partition which is uniquely assigned to a second host computer of the plurality of host computers.

26. The system of Claim 19, wherein the host computers and the storage servers implement an authentication protocol in which a storage server authenticates a host computer before allowing the host computer to perform input/output operations.

27. The system of Claim 19, wherein the host computers and the storage servers implement a discovery protocol in which a storage server notifies a host computer of partitions assigned to the host computer.

28. A method of performing input/output operations, comprising:

establishing first and second TCP/IP connections between a host computer and a block-level storage server over one or more computer networks;

performing a first input/output operation over the first TCP/IP connection concurrently with performing a second input/output operation over the second TCP/IP connection, each of said input/output operations comprising a transfer of input/output data between the host computer and the storage server; and

maintaining the first and second TCP/IP connections in a persistent state such that each TCP/IP connection may be used to perform additional input/output operations.

29. The method as in Claim 28, wherein the first and second TCP/IP connections are established over separate computer networks.

30. The method as in Claim 28, wherein the first input/output operation is a first I/O request issued from a first application running on the host computer, and the second input/output operation is a second I/O request issued from a second application running on the host computer.

31. The method as in Claim 28, wherein the first and second input/output operations are constituent operations of an I/O request issued by a process running on the host computer, whereby the I/O request is executed in parallel over multiple TCP/IP connections.

32. The method as in Claim 28, further comprising establishing a third TCP/IP connection between the host computer and the storage server, and using the third TCP/IP connection to perform an authentication sequence in which the storage server authenticates the host computer.

33. The method as in Claim 32, further comprising conveying access information over the third TCP/IP connection from the storage server to the host computer, said access information specifying access rights uniquely assigned to the host computer.

34. A method of executing an input/output (I/O) request received from a user-level process running on a host computer, comprising:

on the host computer, dividing the I/O request into multiple constituent I/O operations; and

performing the multiple constituent I/O operations in parallel over multiple, respective logical network connections between the host computer and a target storage server such that I/O data is transferred between the host computer and the storage server over each of the logical network connections.

35. The method of Claim 34, wherein each logical network connection is a socket connection.

36. The method of Claim 34, wherein each logical network connection is a TCP/IP socket connection.

37. The method of Claim 34, wherein at least one of the logical network connections is over a general-purpose computer network.

38-53: (Canceled)

54. The network storage system as in Claim 1, wherein the host computer and the storage server communicate with each other over each of the logical connections using a TCP/IP protocol.

55. The system of Claim 19, wherein a first host computer of said plurality of host computers is programmed to open first and second socket connections over the network to a first storage server of said plurality of storage servers, and to perform a first input/output operation over the first socket connection while performing a second input/output operation over the second socket connection.

56. The system of Claim 19, wherein the host computers and storage servers are programmed to perform said input/output operations via TCP/IP communications over said socket connections.

57. The method of Claim 28, wherein the method comprises establishing and maintaining the first and second TCP/IP connections over an Ethernet network.

58. The method of Claim 34, wherein performing the multiple constituent I/O operations comprises the host computer communicating with the target storage server over the multiple logical network connections using a TCP/IP protocol.

59. The method of Claim 34, wherein the multiple constituent I/O operations are performed over an Ethernet network to which the host computer and the storage server are connected.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None